Discerning Differences: Synthetic Text and Human Text

Brian Dang
Department of Information
and Computer Sciences
University of Hawaii at Manoa
Honolulu, Hawaii
Email: bdd@hawaii.edu

Hubert Liang
Department of Information
and Computer Sciences
University of Hawaii at Manoa
Honolulu, Hawaii
Email: hubertl@hawaii.edu

Raul Harnasch
MIT Lincoln Laboratory
AI Technology and Systems
Lexington, Massachusetts
Email: raul.harnasch@ll.mit.edu

***Abstract*—As generative artificial intelligence (GenAI), exemplified by models like ChatGPT, gains prominence, it concurrently raises concerns about its potential misuse. This study examines the effectiveness of synthetic text detection methods by replicating and expanding upon the methodology from Shijaku and Canhasi's "ChatGPT Generated Text Detection" (2023). Utilizing their original dataset alongside the newly introduced DeepfakeTextDetect dataset, we applied traditional machine learning models, such as XGBoost and Logistic Regression, to compare the efficacy of custom lexical features against standard TF-IDF vectorization. Our findings question the superiority of lexical feature sets over simpler statistical methods, as no significant performance enhancements were observed. Furthermore, this study explores the generalizability of these methods across different text datasets, revealing considerable variances in model performance when applied to unseen data. This research underscores the need for ongoing refinement of text detection methodologies to keep pace with advancing GenAI capabilities, highlighting the potential of simpler, more robust feature extraction techniques in enhancing synthetic text detection.**

***Index Terms*—synthetic text detection, generative artificial intelligence, natural language generation, dataset generalization, cross validation, princicpal component analysis, logistic regression, linear discriminant analysis**

I. Introduction

With the implementation of ChatGPT and its rising popularity, Generative Artificial Intelligence (GenAI) has taken the forefront of Artificial Intelligence. Due to its ability to produce human-like responses, concerns regarding its misuse have also risen. With the growth of Natural Language Generation (NLG) models, AI detection tools must also stay relevant to prevent the potential threats of misuse of synthetic text generation [1]. Detecting and discerning between human and synthetic text is pivotal for maintaining authenticity and security in digital communication, with broad implications across many facets of digital life (social media, advertising, scams, etc.) [2]. Despite the concerns, knowledge and access to the NLG models are becoming widespread. As text generation popularity rises, the potential misuse of the tool also increases. To combat this, tools such as GPTZero have been created in order to help differentiate between human generated and synthetic text with some success [3].

In our research, we examine the text detection methods as proposed by Shijaku and Canhasi in their 2023 paper [4], recreating their methods and expanding upon their experimentations. The authors created a dataset[1] containing human-written texts from the Test of English as a Foreign Language (TOEFL) exam and machine-generated models from the ChatGPT model. Using the author's open source dataset, we recreate their 244 lexical features as described by the paper, along with an off-the-shelf implementation of the flesch-kincaid grade level measure.

We aim to replicate the methodology outlined in the original paper in order to extend our evaluation to a broader dataset as well as estimate the generalization of the author's approach This broader evaluation will help assess if custom, statistics-based feature extraction is viable in light of more standard approaches Doing so, we aim to achieve a step further in the ongoing development of reliable synthetic text detection methods.

A. Datasets

1) TOFEL: As mentioned, to recreate the findings in the "ChatGPT Generated Text Detection" paper, we use the TOEFL essays dataset. TOEFL is a widely used standardized test to measure the English language ability of non-native speakers. The dataset contains 252 essays consisting of 126 TOEFL human essays and 126 ChatGPT essays based on questions used on the TOEFL exams.

2) DeepfakeTextdetect: The Deep Fake Text Detection dataset will be utilized to assess both the potential of transfer learning and generalizability of lexical features, serving as a benchmark for machine learning models and evaluating the proficiency of synthetic text. The dataset consists of different text content, such as news article writing, story generation, scientific writing, etc. It contains 447,674 human and machine-generated texts. The dataset consists of 93,318 human texts and 225,753 machine-generated texts. For our experiment, we extracted 20,000 texts: 10,000 human, and 10,000 machine generated text.

[1] https://github.com/rexshijaku/chatgpt-generated-text-detection-corpus/tree/main

II. Approach

Our methodology was to reconstruct the author's custom lexical features, but were only able to successfully ascertain and implement 239 of the original 244 outlined in the paper (see Figure 4). In the process, several discrepancies were noted, which we hypothesize accounts for the five missing features.

Firstly, we had contacted the authors of the original paper and they noted that the 36-character unigram frequencies found in the lexical features represents the 36 letters of the Albanian language. These features differ from the English-based features we extracted accordingly. The original authors had also noted that they mistakenly used code from another study and applied it to English text as shown in their paper "ChatGPT Generated Text Detection". So in our study we only used 26-character unigram frequencies representing the 26 letters of the English language. Additionally, the original study considered 30 special characters, which were not enumerated in the paper, whereas we identified 32 special characters, derived from the symbols found in the English dataset. We also incorporated the Flesch-Kincaid grade level score into our lexical features after noting its use as a feature in their figures, despite it not being included in the lexical descriptions in Figure 4 or referenced otherwise.

After recreating the feature extraction using the author's dataset and assessing the features, we then employed two predictive models: XGBoost (as the authors did) along with Logistic Regression for a direct comparison between the term frequency inverse-document frequency (TF-IDF) feature extraction and the original feature extraction. Logistic regression was chosen as a baseline model to compare to XGBoost due to its simplicity and easy implementation. We also noted that the values found within the lexical feature dataset are statistically based and not complex, so using a Logistic Regression model could potentially be favorable in classification [5].

In the original paper, TF-IDF serves as the benchmark feature set for comparison with the lexical feature set. TF-IDF is considered to be a common approach for text vectorization due to its scalable out-of-the-box implementation. Essentially, TF-IDF evaluates word importance based on word frequency, making it a statistical based method. Leveraging Logistic Regression, it can further enhance its effectiveness.

Mimicking the paper's testing, we evaluated XGBoost and Logistic Regression opting to further utilize 5 K-Fold cross-validation to bolster performance. To also evaluate the viability of adding Flesch-Kincaid Grade Level, we conducted tests with and without Flesch-Kincaid as part of the lexical features. The models were then evaluated on a new dataset, DeepfakeTextDetect, (also using 5 K-Fold cross-validation) to validate the effectiveness of the proposed feature set approach and testing feature generalizability.

Our testing also included pre-training the selected models on the author's data and testing performance on the Deepfake-TextDetect, and vice versa.

Furthermore, we also applied Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA) on the models trained on the TOEFL lexical feature dataset to identify any redundant features and potentially reduce dimensionality. After obtaining any standout features, we will evaluate the features using XGBoost and Logistic Regression. We want to determine if reducing the number of redundant features leads to an evaluation that performs just as well as our experiment recreation. Essentially eliminating unnecessary features.

We chose PCA due to its strengths in dimensionality reduction, and its ability for exploratory analysis on data. PCA identifies usages of redundant or less important values by identifying features with the highest variance for each principal component. Similar to PCA, LDA is also a popular dimensionality reduction tool. It differs from PCA by focusing on maximizing the separability among features.These methods also come with advantages and disadvantages. PCA is able to reduce noise within data and reveal hidden patterns, however it can also lose important features that have low variance but high relevance. On the other hand, LDA can improve model performance by maximizing class separation [6]. Yet, LDA could also potentially produce biased results and overfit to prominent features. So we employ both methods to help us identify as many relevant features in the data. This is to maximize the advantages from both methods and to minimize the potential disadvantages.

III. Evaluation

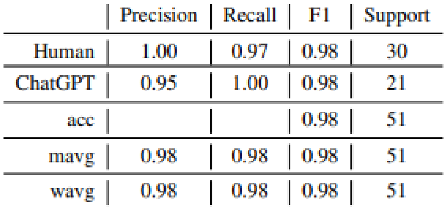
Our evaluation focused on the accuracy of the models for both the TOEFL and (the new) dataset. Looking at Table I, initial results showed that Logistic Regression on TF-IDF features performed comparably to XGBoost, questioning the need for more complex feature engineering. We were also unable to recreate either their lexical results nor the TF-IDF results. We included and excluded flesch-kincade (as noted), to no noticeable effect. Looking at Figure 1 & 2, XGBoost had accuracy scores of 98.00% and 96.00% associated with custom and TF-IDF feature extraction methods respectively, while we could only achieve 91.28% and 92.05% accuracy respectively (Table I).

TABLE I: Results From Recreation

Lexical Features w/ Flesch-Kincaid		
	XGBoost	Logistic Regression
Accuracy	91.28%	90.48%
Precision	91.81%	92.40%
Recall	91.26%	88.18%
F1	91.30%	90.11%

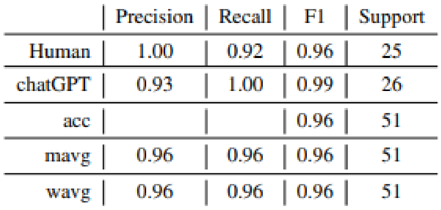
Lexical Features w/o Flesch-Kincaid		
	XGBoost	Logistic Regression
Accuracy	90.47%	90.48%
Precision	92.14%	92.40%
Recall	88.89%	88.18%
F1	91.30%	90.11%

TF-IDF		
	XGBoost	Logistic Regression
Accuracy	92.05%	95.22%
Precision	93.55%	99.13%
Recall	90.40%	91.26%
F1	91.86%	94.90%



	Precision	Recall	F1	Support
Human	1.00	0.97	0.98	30
ChatGPT	0.95	1.00	0.98	21
acc			0.98	51
mavg	0.98	0.98	0.98	51
wavg	0.98	0.98	0.98	51

Fig. 1: TF-IDF results from ”ChatGPT Generated Text Detection”



	Precision	Recall	F1	Support
Human	1.00	0.92	0.96	25
chatGPT	0.93	1.00	0.99	26
acc			0.96	51
mavg	0.96	0.96	0.96	51
wavg	0.96	0.96	0.96	51

Fig. 2: Lexical feature results from ”ChatGPT Generated Text Detection”

Implementing the original paper’s approach onto a new dataset revealed its ability to generalize to different datasets (See Table II). In this evaluation we trained and tested on the DeepfakeTextDetect dataset. First, looking at our lexical feature evaluation, XGBoost scores an accuracy of 94.29%. When trained and tested on TF-IDF, XGBoost scores an accuracy of 90.42%. Logistic Regression on the other hand, drops in performance compared to the original testing results, as seen in Table I.

TABLE II: Approach Generalization

Lexical Features w/ Flesch-Kincaid		
	XGBoost	Logistic Regression
Accuracy	94.29%	83.49%
Precision	93.65%	84.66%
Recall	96.94%	88.10%
F1	95.27%	86.34%

Lexical Features w/o Flesch-Kincaid		
	XGBoost	Logistic Regression
Accuracy	94.18%	83.52%
Precision	93.75%	84.70%
Recall	96.62%	88.12%
F1	95.16%	86.37%

TF-IDF		
	XGBoost	Logistic Regression
Accuracy	90.42%	88.58%
Precision	90.90%	95.49%
Recall	89.88%	81.00%
F1	90.36%	87.61%

In our pre-trained evaluation, we initially trained our models on the TOEFL dataset and then assessed their performance on the DeepfakeTextDetect dataset. Our findings, illustrated in Table III, indicate a decline in model performance when evaluated with new data. XGBoost achieves an accuracy of 62.53%, while Logistic Regression attains 53.74%. In the next step of our evaluation, we reversed the pre-training process by training our models on the DeepfakeTextDetect dataset and evaluating them on the TOEFL dataset. Contrasting these results with our initial evaluation, we observed a deterioration in accuracy scores. XGBoost decreased to 44.05% accuracy, and Logistic Regression decreased to 46.43%, as depicted in the table.

TABLE III: Reciprocal Evaluations: w/ Flesch-Kincaid

Train on TOEFL, Test on DeepfakeTextDetect		
	XGBoost	Logistic Regression
Accuracy	62.53%	53.74%
Precision	70.47%	52.83%
Recall	67.64%	63.10%
F1	69.03%	57.51%

Train on DeepfakeTextDetect, Test on TOEFL		
	XGBoost	Logistic Regression
Accuracy	44.05%	46.43%
Precision	39.68%	57.14%
Recall	43.48%	47.06%
F1	41.49%	51.61%

TABLE IV: Reciprocal Evaluation: w/o Flesch-Kincaid

Train on TOEFL, Test on DeepfakeTextDetect		
	XGBoost	Logistic Regression
Accuracy	62.16%	53.24%
Precision	72.45%	51.78%
Recall	66.62%	62.78%
F1	69.41%	56.75%

Train on DeepfakeTextDetect, Test on TOEFL		
	XGBoost	Logistic Regression
Accuracy	50.40%	46.83%
Precision	45.24%	55.56%
Recall	50.44%	47.30%
F1	47.70%	51.09%

A. *Data/Feature Exploration*

Based on our data analysis, our initial hypothesis regarding unnecessary custom features appears to be incorrect. PCA didn’t uncover any distinct features, suggesting that our assumptions might have been off base. However, LDA highlighted that ”total character count” and ”word count” emerged as significant features. Remarkably, a model trained solely on these two attributes achieved an impressive accuracy of 71.11%. This underscores further exploration into feature relevance and potential contribution to model performance.

We have also explored the potential importance of Flesch-Kincaid Grade Level score as a temporal feature across the TOEFL essays. This was done by splitting each essay into 3 segments and graphing the average across all essays (human and ChatGPT generated). Showcased in Figure 3, we notice a trend amongst the Human segments; On average the TOEFL human essays tend to drop in score by the 2nd segment. In contrast, ChatGPT generated essay scores increased over each segment.

After observing these differences, we explored the data by creating a simple rule-based classifier to differentiate the essays based on their Flesch-Kincaid segment score. The classifier followed the following rules: If segment 1 is less than segment 2 then the essay is classified as ChatGPT, otherwise the essay is classified as Human. Using this method, we achieved an accuracy of 69.79% (Table V). Although it can be seen in our previous evaluations that the addition of Flesch-Kincaid grade level did not considerably affect scoring, Figure 3 suggests promising avenues for future exploration.

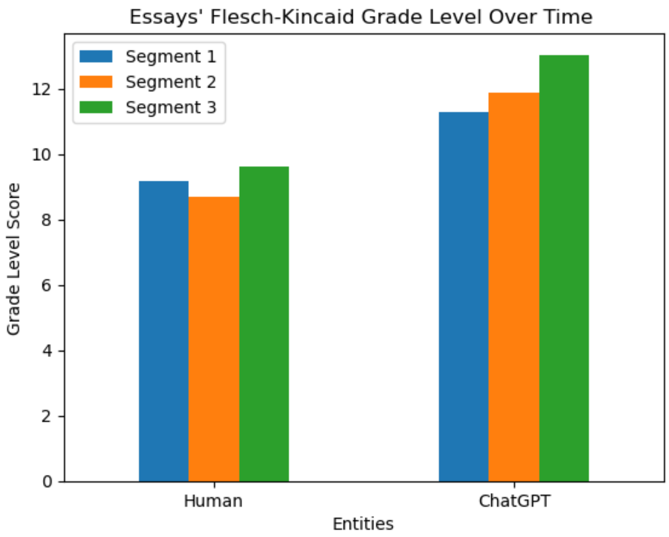


Fig. 3: Flesch-Kincaid Grade Level Over Time

TABLE V: Flesch-Kincaid Segmentation

Classification by segment length	
Accuracy	69.79%
Precision	67.16%
Recall	86.54%
F1	75.63%

IV. Discussion

In our replication of the results from the original paper [4], we tested the original TOEFL dataset using the Term Frequency-Inverse Document Frequency (TF-IDF) vectorization and found that it outperformed the feature extraction approach advocated by the author. This finding contradicts the authors' claim that lexical feature extractions were superior to TF-IDF, as we observed no significant difference in recall, F1, and accuracy scores that would clearly indicate lexical feature extraction performing over TF-IDF (Figure 1). Variations in model performance may be explainable by differences in text preprocessing, such as stemming and the application of other normalization techniques. Furthermore, we observed no noticeable difference in performance between the XGBoost and Logistic Regression models, which challenges the need for heightened complexity when simpler models provide comparable outcomes.

Despite the earlier findings, it's important to note a significant caveat: the custom feature approach demonstrated better generalization to other datasets when models were not pre-trained. Moving on to model generalization, we noticed an asymmetric performance when swapping the training and testing datasets. Training on the new dataset and testing on the original (TOEFL) dataset yielded much worse results compared to the reverse (Table III). This asymmetry may be indicative of overfitting, particularly as the smaller TOEFL dataset, while potentially performing "favorably" on a broader type of English text such as news, lacks the diversity to capture the niche details present in larger, more varied datasets. Conversely, the larger dataset, with its extensive characteristics, struggles to identify the specific features of the TOEFL data, highlighting challenges in model robustness across different data sources. Our evaluation suggests that the lexical feature extraction approach, despite inconsistencies noted in the original study's findings, may still be more generalizable than TF-IDF when tested against the larger datasets (Table II).

V. Future Work

Future research in the area of text generation detection could benefit from exploring alternative models beyond XGBoost and Logistic Regression to holistically identify which models may be best suited for the task of identifying human-vs-synthetic text. Moreover, a deeper investigation into feature selection among the TOEFL dataset could refine the detection process. As stated prior, there are features that are present which do not contribute meaningfully to the classification, most likely as a result of an overfitting.

Additionally, exploring the role of the Flesch-Kincaid readability score in conjunction with other linguistic features such as burstiness and perplexity may result in insightful findings, as trends in these metrics have been observed in conjunction with the text (Figure 3). Lastly, we performed K-Fold cross-validation in our study, which showed no discernible effect, but this could be improved on with adjustments to hyperparameter optimization, to which the original authors did not fully explore.

VI. Conclusion

In conclusion, our study has highlighted the complexities involved in distinguishing between human-written and machine-generated text. While the original authors found success with lexical features and XGBoost, our findings suggest that simpler models and alternative feature extraction methods, such as TF-IDF, can perform equally well or even better in certain contexts.

Our analysis highlights the challenges of applying models across varied datasets, demonstrating that while the lexical feature approach may be more generalizable, TF-IDF remains a robust method for text classification. Furthermore, our results indicate that custom, statistics-based features and other linguistic measures, such as Flesch-Kincaid readability scores, burstiness, and perplexity, merit further investigation to enhance detection capabilities.

Overall, the study underscores the importance of carefully selecting and testing feature extraction methods and classification models based on the specific requirements of the task at hand. Moving forward, further exploration into more advanced models and more refined feature selection processes may ultimately aid in more accurately and reliably detecting synthetic text.

VII. Acknowledgement

Special thanks to Raul Harnasch for mentorship, guidance, and direction.

References

[1] B. C. Gabbiadini Alessandro, Ognibene Dimitri and M. Anna, “The emotional impact of generative ai: negative emotions and perception of threat,” *Behaviour & Information Technology*, vol. 0, no. 0, pp. 1–18, 2024.

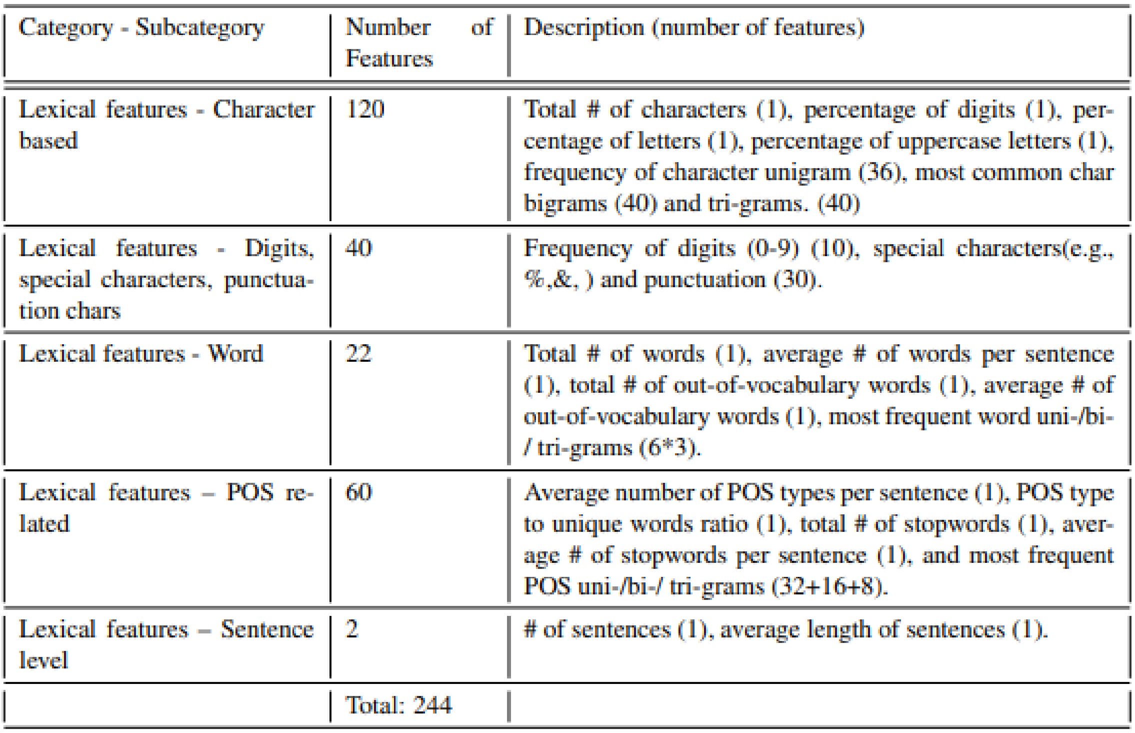
[2] N. A. Nawaz, K. Ishaq, U. Farooq, A. Khalil, S. Rasheed, A. Abid, and F. Rosdi, “A comprehensive review of security threats and solutions for the online social networks industry,” Jan 2023.

[3] J. H. Kirchner, L. Ahmad, S. Aaronson, and J. Leike, “New ai classifier for indicating ai-written text.”

[4] R. Shijaku and E. Canhasi, “Chatgpt generated text detection.”

[5] M. K. Enduri, A. R. Sangi, S. Anamalamudi, R. C. B. Manikanta, K. Y. Reddy, P. L. Yeswanth, S. K. S. Reddy, and G. A. Karthikeya, “Comparative study on sentimental analysis using machine learning techniques.”

[6] R. Pramoditha, “Lda is more effective than pca for dimensionality reduction in classification datasets,” Jan 2023.



Category - Subcategory	Number of Features	Description (number of features)
Lexical features - Character based	120	Total # of characters (1), percentage of digits (1), percentage of letters (1), percentage of uppercase letters (1), frequency of character unigram (36), most common char bigrams (40) and tri-grams. (40)
Lexical features - Digits, special characters, punctuation chars	40	Frequency of digits (0-9) (10), special characters(e.g., %,&,) and punctuation (30).
Lexical features - Word	22	Total # of words (1), average # of words per sentence (1), total # of out-of-vocabulary words (1), average # of out-of-vocabulary words (1), most frequent word uni-/bi-/ tri-grams (6*3).
Lexical features – POS related	60	Average number of POS types per sentence (1), POS type to unique words ratio (1), total # of stopwords (1), average # of stopwords per sentence (1), and most frequent POS uni-/bi-/ tri-grams (32+16+8).
Lexical features – Sentence level	2	# of sentences (1), average length of sentences (1).
	Total: 244	

Fig. 4: Lexical features from ”ChatGPT Generated Text Detection”